**BAB IV**

**ANALISA DAN PEMBAHASAN**

1. **Analisa Sistem**
   * 1. **Identifikasi Masalah**

Untuk memperjelas permasalahan yang timbul maka penulis akan mencoba memformulasikan ke dalam bentuk perumusan masalah. Berikut ini akan diuraikan tentang hal-hal yang menjadi penyebab timbulnya masalah yang dapat menimbulkan kebutuhan untuk pembuatan system baru :

1. Pada proses pembuatan surat pengantar, warga masih menggunakan system manual yaitu warga dating ke ketua RT kemudian mengisi form surat pengantar selanjutnya di mintakan persetujuan ke ketua RW setelah selesai baaru surat permohonan di bawa ke kelurahan untuk memohon apa yang diajukan di dalam surat pengantar, sehingga menyebabkan kurang cepat dan memerlukan waktu yang lama.
2. Masalah dating jika ketua RT atau RW tidak ada dirumah maka warga harus datang kembali keesokan harinya sehingga untuk permohonan pembuatan surat pengantar harus tertunda *.*
3. **Analisis Kebutuhan Sistem Fungsional**

Kebutuhan fungsional (*functional requirement)* adalah jenis kebutuhan yang berisi proses – proses apa saja yang dapat dibutuhkan oleh system. Kebutuhan fungsional juga berisi informasi – informasi apa saja yang harus ada dan di hasilkan oleh system. Berikut adalah kebutuhan fungsional dari Aplikasi Surat pengantar RT RW berbasis Web :

* + - 1. Client / Warga
  1. Sistem bisa membuat surat pengantar.
  2. Sistem dapat menampilkan surat pengantar sudah di validasi atau belum oleh RT, RW dan informasi – informasi selanjutnya.
     + 1. Admin (Ketua RT, Ketua RW, Kelurahan)
          1. Sistem dapat melakukan login.
          2. Sistem dapat melakukan pengolahan data penduduk seperti input penduduk baru, edit, hapus.
          3. Sistem dapat melakukan validasi sesuai user.
          4. Sistem dapat mencetak hasil surat pengantar.

1. **Analisa Kebutuhan Sistem**
2. **Analisa Kebutuhan Hardware**

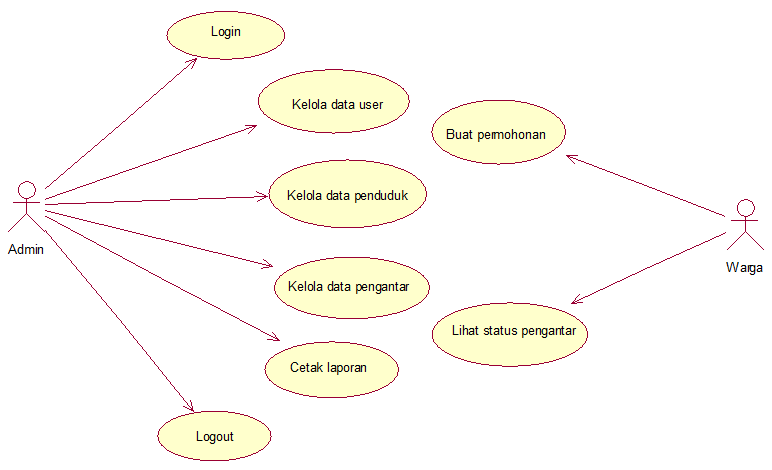
Berikut adalah hardware dengan sepesifikasi minimum yang dapat digunakan penulis untuk pembuatan aplikasi surat pengantar Rt / Rw berbasis web adalah sebagai berikut :

1. *Processor intel core i3* atau lebih tinggi.
2. Monitor 14” resolusi layar 1024x768 pixels.
3. Hardisk 500 GB atau lebih tinggi.
4. RAM 4GB atau lebih tinggi
5. Keyboard
6. Mouse
7. Printer
8. Modem untuk koneksi internet.
9. **Analisa Kebutuhan Software**

Perangkat lunak yang dibutuhkan dalan pembuatan aplikasi surat pengantar Rt / Rw berbasis web sebagai berikut:

1. Sistem operasi windows 7 atau versi yang lebih tinggi ( windows 10 pro lebih direkomendasikan ).
2. Xampp V3.2.1.
3. Web browser (Mozila firefox, Crome, Opera, dll).
4. Notepad++ atau sublime text 3.
5. Framework CodeIgniter 3.1.6.
6. **Perancangan Sistem UML (*****Unified Modeling Language).***
   1. ***Use Case Diagram***

*Use case* diagrammenggambarkan fungsionalitas sistem secara global. Berikut ini adalah *use case* yang berjalan di bagian



Gambar 4.1*Use Case Diagram*

**Sekenario Use Case melakukan Login**

Nama *Use Case* : Melakukan Login

Aktor : Admin

Deskripsi : Proses login admin merupakan proses

Masuk agar diijinkan mengakses dengan

memasukkan username dan password

yang telah divalidasi

*Pre – Condition* : Admin belum login ke dalam sistem

*Post – Condition* : Admin berhasil masuk ke dalam sistem

Dan dapat mengaksesn

Tabel 4.1 Skenario Melakukan Login

|  |  |
| --- | --- |
| **Aksi aktor** | **Reaksi Sistem** |
| **Alur untuk melakukan login** | |
| 1. Admin masuk system dan memilih menu login |  |
|  | 1. Sistem menampilkan halaman login |
| 1. Admin mengisi username dan password |  |
|  | 1. Sistem akan melakukan validasi data username dan password |

**Skenario Use Case Mengelola Data Admin / Petugas**

Nama *use case* : Mengelola daftar data admin

Aktor : admin

Deskripsi : Kegiatan untuk mendaftarkan data admin

baru, mengubah data admin yang sudah ada, ataupun menghapus anggota yang lama.

*Pre-condition* : Admin harus berada di menu utama dan

sudah *login*.

*Post*-*condition* : Data informasi baru tersimpan.

Tabel 4.2 Skenario Mengelola Data Admin / Petugas

|  |  |
| --- | --- |
| **Aksi Aktor** | **Reaksi Sistem** |
| **Menambah data** | |
| 1. Pilih menu list petugas pada menu daftar petugas |  |
|  | 1. Menampilkan halaman list petugas / admin |
| 1. Pilih tombol “Tambah petugas” untuk menambah petugas |  |
|  | 1. Menampilkan form modal tambah petugas |
| 1. Memasukkan data petugas baru seperti nama, alamat, phone, email, jabatan. |  |
| 1. Menekan tombol “Simpan” |  |
|  | 1. Mengecek valid tidaknya data masukan dan mengidentifikasi apakah petugas sudah pernah dimasukan atau belum |
|  | 1. Jika data petugas yang dimasukan valid, maka data akan disimpan di database |
| 1. Petugas melalukan pengecekan data. |  |
| **Memperbarui data admin / petugas** | |
| 1. Pada menu list data Petugas pilih Ikon “Edit” untuk memperbarui data |  |
|  | 1. Menampilkan data-data petugas yang telah tersimpan. |
| 1. Memperbarui data seperti nama, alamat, phone, email, jabatan |  |
| 1. Klik tombol “Simpan” untuk menyimpan pembaharuan |  |
|  | 1. Mengecek valid tidaknya data yang diperbaharui |
|  | 1. Jika data yang dimasukan valid maka data anggota yang diperbaharui akan disimpan didatabase |
| **Menghapus data petugas** | |
| 1. Pada menu List Data Petugas pilih Ikon “Hapus” untuk menghapus data |  |
|  | 1. Menampilkan popup atau alert peringatan apakah ingin benar menghapus data |
| 1. Klik “Hapus” |  |
|  | 1. Data berhasil dihapus dari database |

* + - 1. **Skenario Use Case Mengelola Data Penduduk**

Nama *use case* : Mengelola data penduduk

Aktor : Admin

Deskripsi : Kegiatan untuk mendaftarkan penduduk

baru ataupun mengubah data informasi penduduk yang sudah ada.

*Pre-condition* : Admin harus berada di menu utama dan

sudah *login*.

*Post*-*condition* : Data informasi penduduk baru tersimpan

Tabel 4.3 Skenario Mengelola Data penduduk

|  |  |
| --- | --- |
| **Aksi Aktor** | **Reaksi Sistem** |
| **Alur untuk menambah data Penduduk baru** | |
| 1. Klik menu data master dan pilih cara input data penduduk jika memilih dengan form atau import excel |  |
|  | 1. Manampilkan form modal input penduduk |
|  | 1. Menampilkan menu import file excel |
| 1. Memasukan data penduduk ke form atau file excel |  |
| 1. Klik Ikon “Simpan” untuk menambah data penduduk baru |  |
|  | 1. Mengecek valid tidaknya data masukan dan mengidentifikasi apakah penduduk sudah pernah dimasukan atau belum |
|  | 1. Jika data penduduk yang dimasukan valid, maka data akan disimpan di database |
| 1. Admin melalukan pengecekan data. |  |
| **Alur untuk memperbaharui data penduduk** | |
| 1. Pada menu list data Penduduk pilih Ikon “Edit” untuk memperbarui data |  |
|  | 1. Menampilkan informasi data penduduk di form modal |
| 1. Menekan Ikon “Edit” |  |
|  | 1. Mengecek valid tidaknya data yang diperbaharui |
|  | 1. Jika data yang dimasukan valid maka akan disimpan didatabase |
|  | 1. Menampilka alert data berhasil diubah |
| **Menghapus data penduduk** | |
| 1. Pada menu List Data Peenduduk pilih Ikon “Hapus” untuk menghapus data |  |
|  | 1. Menempilkan alert apakah ingin menghapus data |
| 1. Menekan Ikon “Hapus” |  |
|  | 1. Data akan dihapus dari database |
|  | 1. Menampilkan alert data penduduk berhasil di hapus |

* + - 1. **Skenario Use Case Mengelola data Pengantar**

Nama *use case* : Mengelola data pengantar

Aktor : Admin

Deskripsi : Proses ini adalah kegiatan untuk mngecek data permohonan pengantar benar – benar sesuai perosedur pembuatan pengantar atau tidak, jika sesuai pengantar bisa di validasi oleh admin dan akan dihapus jika tidak sesuai.

*Pre-condition* : Admin sudah harus berada di menu validas pengantar dan sudah login.

*Post*-*condition* : Data permohonan pengantar baru tersimpan.

Tabel 4.4 Skenario Mengelola data pengantar

|  |  |
| --- | --- |
| **Aksi Aktor** | **Reaksi Sistem** |
| **Alur untuk mengelola data pengantar** | |
| 1. Klik menu validasi pengantar pada menu validasi |  |
|  | 1. Menampilkan daftar data pengantar baru |
| 1. Klik tombol “validasi” untuk memvalidasi pengantar yang baru |  |
|  | 1. Menampilkan halaman form validasi |
| 1. Memasukkan tangal berlaku pengantar dan isikan keterangan validasi kemudian pilih validasi |  |
| 1. Menekan tombol “Validasi” |  |
|  | 1. Mengecek data yang di validasi kemudian data akan hilang dan di tampilkan di list kelurahan |
| **Menampilkan detail penduduk yang membuat pengantar** | |
| 1. Klik menu validasi pengantar pada menu validasi |  |
|  | 1. Menampilkan daftar data pengantar baru |
| 1. Klik tombol “detail” untuk menampilkan detail penduduk yang telah membuat pengantar |  |
|  | 1. Menampilkan halaman form detail penduduk |
| 1. Menekan tombol “keluar” | 1. Form keluar dan kembali ke daftar data pengantar |
| **Menghapus data pengantar** | |
| 1. Pada menu List Data Peenduduk pilih Ikon “Hapus” untuk menghapus data |  |
|  | 1. Menempilkan alert apakah ingin menghapus data |
| 1. Menekan Ikon “Hapus” | 1. Data akan dihapus dari database |
|  | 1. Menampilkan alert data penduduk berhasil di hapus |

* + - 1. **Skenario Cetak Laporan**

Nama *use case* : Mencetak laporan

Aktor : Admin

Deskripsi : Proses ini adalah kegiatan untuk mencetak laporan pengantar dari RT, RW atau kelurahan jika dibutuhkan untuk arsip.

*Pre-condition* : Admin sudah harus berada di menu laporan dan sudah *login.*

*Post*-*condition* : Data laporan pengantar baru tercetak.

Tabel 4.5 Skenario Mencetak Laporan

|  |  |
| --- | --- |
| **Alur untuk cetak data Pengantar** | |
| **Aksi Aktor** | **Reaksi Sistem** |
| 1. Memilih menu cetak laporan yang berada pada menu cetak |  |
|  | 1. Menampilkan form pencarian data yang ingin di cetak |
| 1. Memasukan NIK atau No pengantar kemudian pilih cari |  |
|  | 1. Memproses data yang di cari |
|  | 1. Menampilkan informasi data pengantar yang akan di cetak |
| 1. Menekan tombol Ikon “Print” |  |
|  | 1. Menampilkan informasi data yang akan dicetak |
| 1. Laporan tercetak |  |

* + - 1. **Skenario *Use Case* melakukan *logout***

Nama *Use* *Case* : Melakukan *Logout*

*Actor* : Admin

Deskripsi : Proses *logout* merupakan proses untuk

keluar dari aplikasi

*Pre – Condition* : Admin sudah login ke dalam aplikasi

*Post – Condition* : Admin berhasil keluar dari aplikasi

Tabel 4.6 Skenario Melakukan *Logout*

|  |  |
| --- | --- |
| **Aksi Aktor** | **Reaksi Sistem** |
| **Alur untuk keluar dari system** | |
| 1. Admin memilih menu logout |  |
|  | 1. Sistem merespon dan memverifikasi |
|  | 1. Sistem melakukan logout |

* + - 1. **Skenario Use Case Buat Permohonan Pengantar**

Nama *use case* : Buat permohonan

Aktor : Warga

Deskripsi : Proses ini adalah kegiatan untuk membuat permohonan surat pengantar baru kedalam aplikasi.

*Pre-condition* : Warga membuka halaman aplikasi *.*

*Post*-*condition* : Warga masuk kehalaman utama aplikasi pembuatan surat pengantar.

Tabel 4.7 Skenario Membuat Permohonan surat pengantar

|  |  |
| --- | --- |
| **Aksi Aktor** | **Reaksi Sistem** |
| **Alur untuk membuat surat pengantar baru** | |
| * + 1. Klik tombol buat pengantar. |  |
|  | 2. Manampilkan form modal input buat pengantar baru |
| 3. masukan NIK maka nama akan muncul otomatis sesuai NIK kemudian isikan keperluan dan lain-lain |  |
| 1. Klik tombol “Buat” untuk membuat surat pengantar baru. |  |
|  | 1. Mengecek valid tidaknya data masukan dan mengidentifikasi apakah penduduk telah terdaftar di wilayah tersebut. |
|  | 1. Muncul popup pengantar berhasil di buat. |

* + - 1. **Skenario Use Case Lihat Status Pengantar**

Nama *use case* : Lihat Status Pengantar

Aktor : Warga

Deskripsi : Proses ini adalah kegiatan untuk melihat status pengantar sudah di validasi atau belum.

*Pre-condition* :Warga membuka halaman aplikasi dan memilih lihat status pengantar *.*

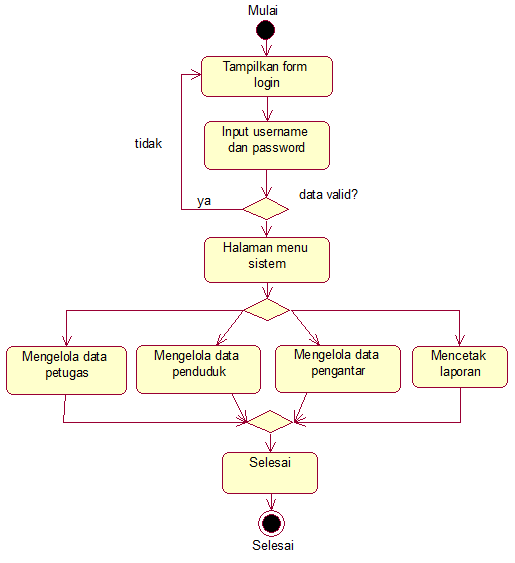
*Post*-*condition* : Warga masuk kehalaman utama aplikasi memasukan NIK kedalam form pencarian .

Tabel 4.8 Skenario melihat setatus pengantar

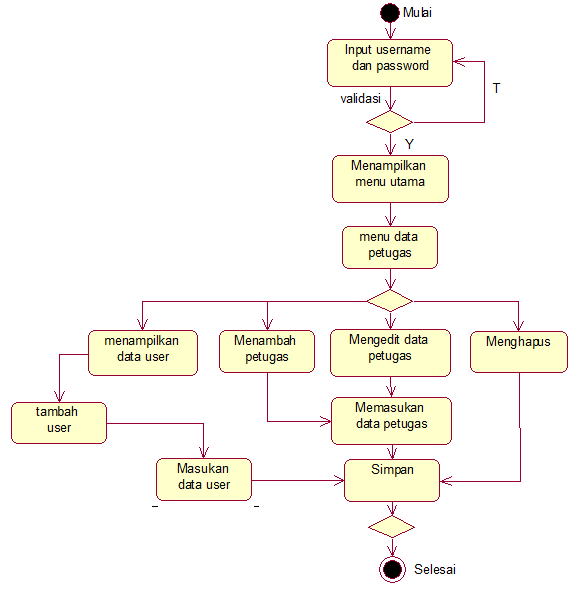
|  |  |
| --- | --- |
| **Aksi Aktor** | **Reaksi Sistem** |
| **Alur untuk melihat setatus surat pengantar** | |
| 1. Klik tombol lihat status pengantar. |  |
|  | 1. Manampilkan form modal cari status pengantar |
| 1. masukan NIK atau nomer surat pengantar |  |
| 1. Klik tombol “Carit” untuk mencari status pengantar. |  |
|  | 1. Mengecek valid tidaknya data masukan dan mengidentifikasi apakah pengantar dengan nomer tersebut ada. |
|  | 1. Menampilkan list status pengantar berdasarkan NIK yang dicari. |

* + 1. ***Activity Diagram***

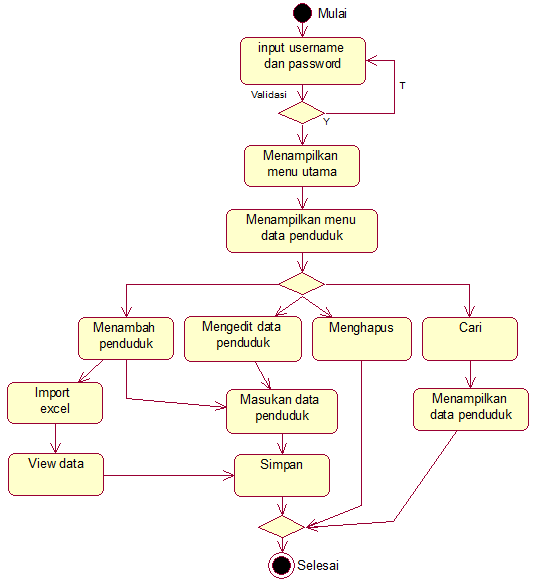
*Activity diagram* menggambarkan rangkaian aliran dari aktivitas yang dapat digunakan untuk mendeskripsikan aktivitas yang dibentuk dalam suatu operasi sehingga dapat juga digunakan untuk aktivitas yang lain seperti *use case* atau interaksi. Aktivitas merupakan bentuk khusus dari keadaan dimana aktivitas menggambarkan kegiatan yang dilakukan. Berikut ini adalah *activity diagram* yang berjalan pada Aplikasi surat pengantar berbasis web di kelurahan krapyak.

1. *Activity Diagram* melakukan Login

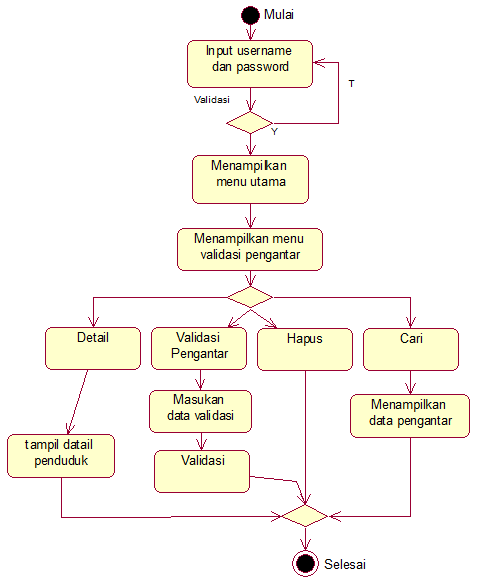
Gambar 4.2 *Activity Diagram* melakukan Login

1. *Activity Diagram* Mengelola data petugas dan user

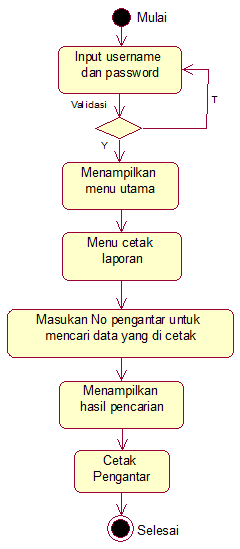
Gambar 4.3 *Activity Diagram* Mengelola data Petugas dan User

1. *Activity Diagram* Mengelola data Penduduk

Gambar 4.4 *Activity Diagram* Mengelola data Penduduk

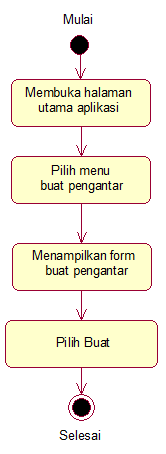
1. *Activity Diagram* Mengelola data Pengantar

Gambar 4.5 *Activity Diagram* Mengelola data Pengantar

1. *Activity Diagram* Cetak Laporan

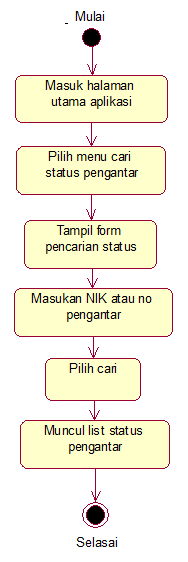
Gambar 4.6 *Activity Diagram* Cetak Laporan

1. *Activity Diagram* Buat Permohonan Pengantar



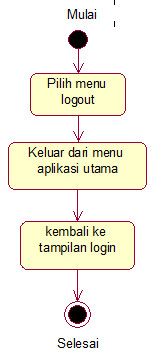
Gambar 4.7 *Activity Diagram* Buat Permohonan Pengantar

1. *Activity Diagram* Lihat Status Pengantar

**

Gambar 4.8 *Activity Diagram* Lihat Status Pengantar

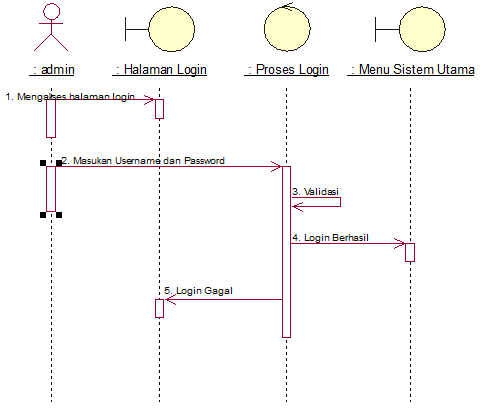
1. *Activity Diagram* Logout



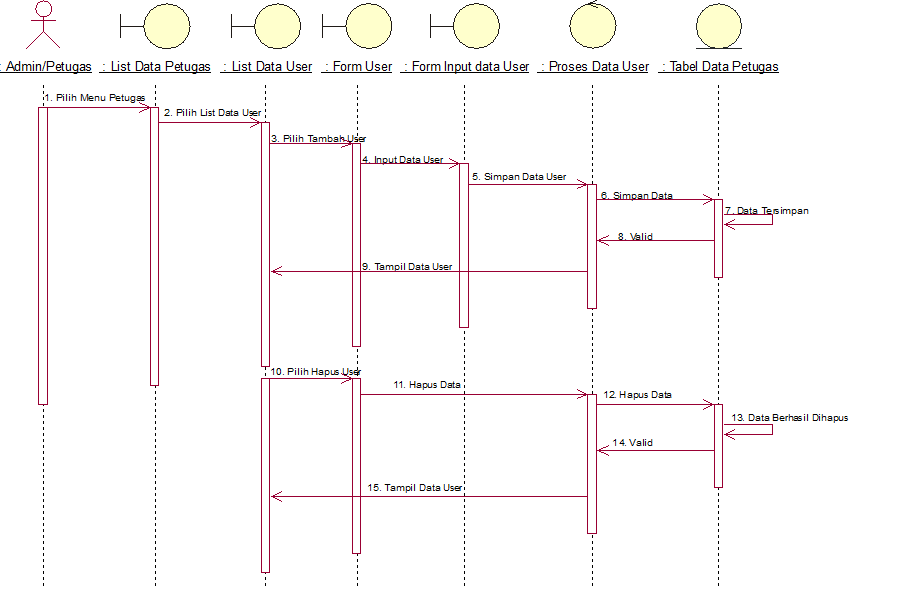
Gambar 4.9 *Activity Diagram* Logout

* + 1. ***Sequence Diagram***

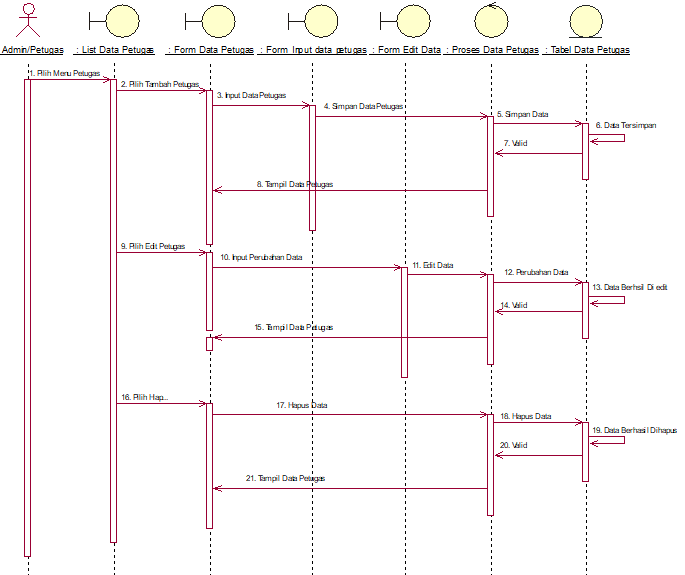
*Sequence diagram* berisi gambaran dari urutan kejadian suatu kegiatan. Menggambarkan kelakuan objek pada use case dengan mendeskripsikan waktu hidup objek dan pesan yang dikirimkan dan diterima antar objek.

1. *******Sequence Diagram* Login

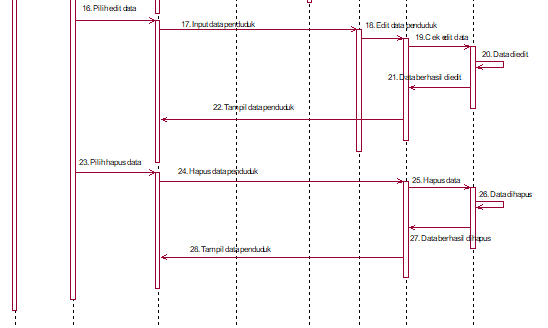
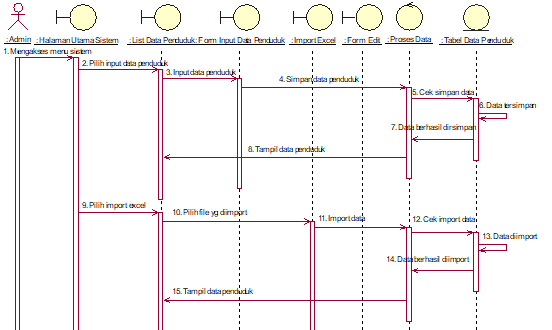
Gambar 4.10 *Sequence Diagram* Login

1. *Sequence Diagram* Mengelola Data User

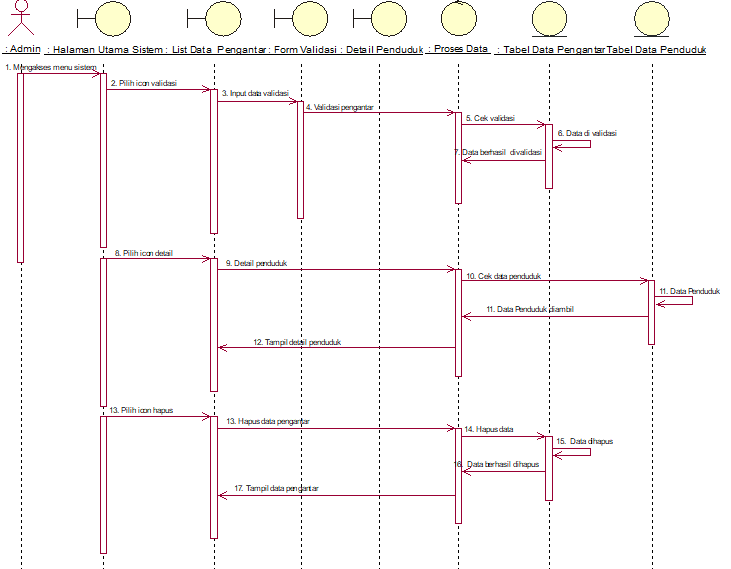
Gambar 4.11 *Sequence Diagram* Mengelola Data User

1. *Sequence Diagram* Mengelola Data Petugas

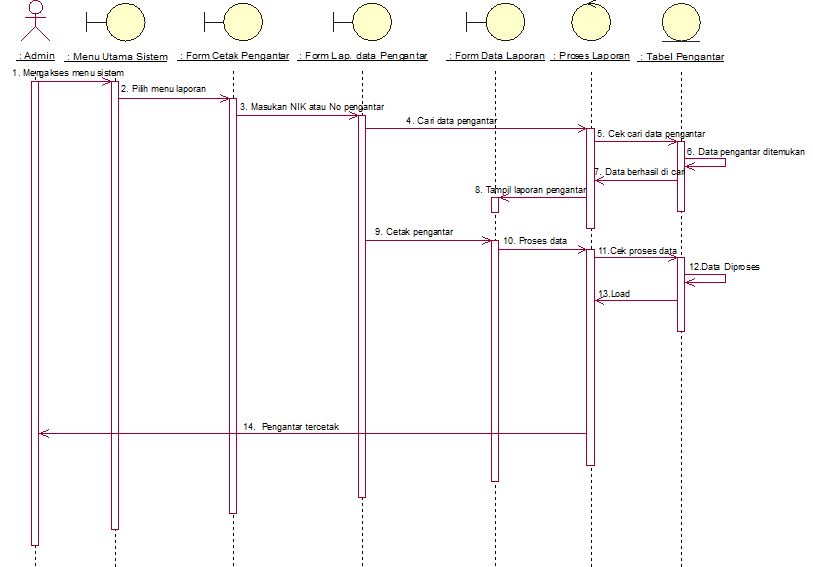
Gambar 4.12 *Sequence Diagram* Mengelola Data Petugas

1. *Sequence Diagram* Mengelola Data Penduduk

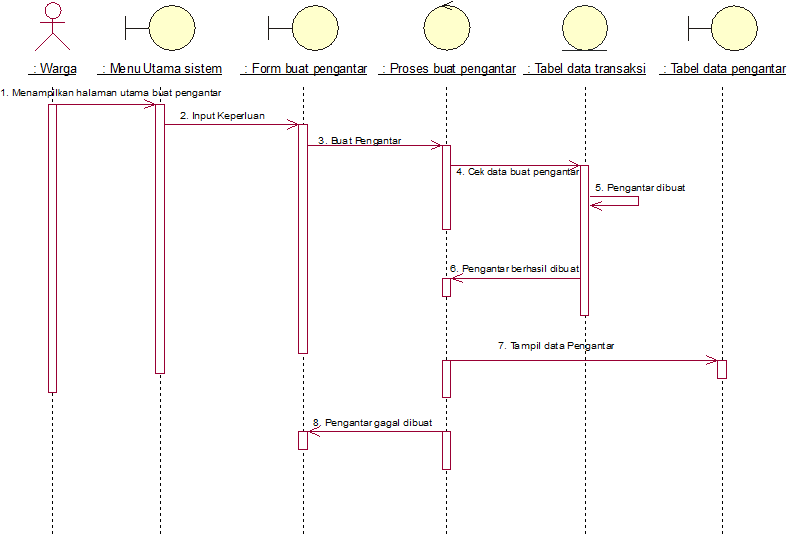
Gambar 4.13 *Sequence Diagram* Mengelola Data Penduduk

1. *Sequence Diagram* Mengelola Data Pengantar

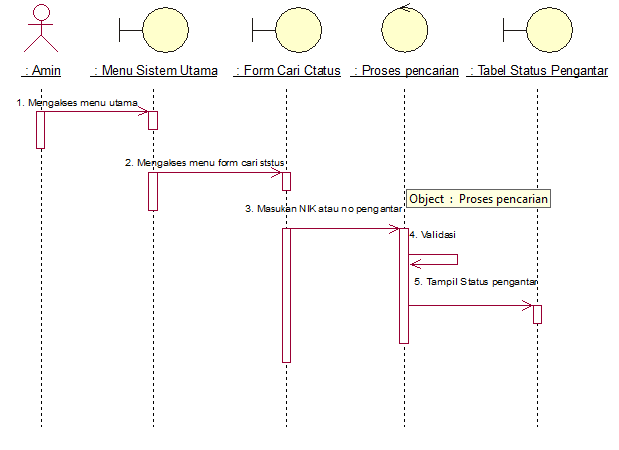
Gambar 4.14 *Sequence Diagram* Mengelola Data Pengantar

1. *Sequence Diagram* Mengelola Cetak Pengantar

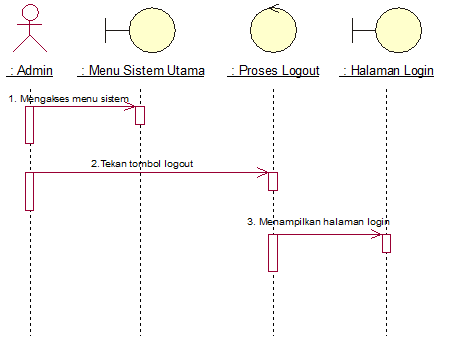
Gambar 4.15 *Sequence Diagram* Cetak Pengantar

1. *Sequence Diagram* Buat Permohonan Pengantar

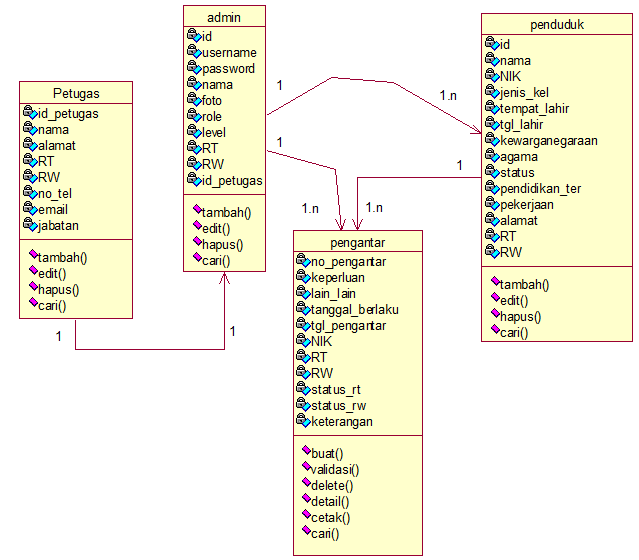
Gambar 4.16 *Sequence Diagram* Buat Pengantar

1. *Sequence Diagram* Lihat Status Pengantar

Gambar 4.17 *Sequence Diagram* Lihat Status Pengantar

1. *Sequence Diagram* Mengelola Logout

Gambar 4.18 *Sequence Diagram* Logout

* + 1. ***Class Diagram***

Gambar 4.19 *Class Diagram*

* 1. **Perancangan Database**

Fungsi dari suatu perancangan *database* adalah untuk menampung beberapa tabel yang digunakan sebagai sumber pengolahan data. Perancangan *database* akan memberikan gambaran mengenai pembuatan tabel didalam *database* beserta atribut – atributnya. Berikut adalah nama – nama tabel yang digunakan beserta field – field yang terdapat pada masing – masing tabel.

* + 1. **Perancangan Tabel Penduduk**

Tabel ini berfungsi untuk menyimpan data penduduk yang nantinya di gunakan untuk pembuatan surat pengantar

Tabel 4.9 Perancangan Tabel Penduduk

|  |  |  |
| --- | --- | --- |
| **Field** | **Type** | **Keterangan** |
| id | Int(6) | *Primary Key*   |  |  | | --- | --- | | (*auto\_increment*) | [Change](http://localhost/phpmyadmin/tbl_structure.php?db=ket_surat&table=penduduk&token=c74aa92b6907c66973e02b8535d1bc83&goto=tbl_structure.php&back=tbl_structure.php&field=id&change_column=1) | |
| NIK | bigint(20) | Foreign key |
| nama | Varchar(30) |  |
| Jenis\_kel | Varchar(11) |  |
| Tempat\_lahir | Varchar(20) |  |
| Tgl\_lahir | Varchar(15) |  |
| Kewarganegaraan | Varchar(10) |  |
| Agama | |  |  | | --- | --- | | Varchar(15) |  | |  |
| Status | Varchar(15) |  |
| Pendidikan\_ter | Varchar(17) |  |
| pekerjaan | Varchar(25) |  |
| alamat | Varchar(50) |  |
| RT | Int(2) |  |
| RW | Int(2) |  |

* + 1. **Perancangan Tabel Pengantar**

Tabel ini berfungsi untuk menyimpan data pembuatan pengantar

Tabel 4.10 Perancangan Tabel Pengantar

|  |  |  |
| --- | --- | --- |
| **Field** | **Type** | **Keterangan** |
| no\_pengantar | Varchar(20) | *Primary Key* |
| keperluan | Text |  |
| lain\_lain | Text |  |
| tangal\_berlaku | Varchar(15) |  |
| tgl\_pengantar | timestamp | |  |  | | --- | --- | | on update CURRENT\_TIMESTAMP |  | |
| NIK | bigint(20) |  |
| RT | Int(2) |  |
| RW | Int(2) |  |
| status\_rt | Int(1) | Foreign key |
| status\_rw | Int(1) | Foreign key |
| Keterangan | Varchar(200) |  |

* + 1. **Perancangan Tabel Admin**

Tabel ini berfungsi untuk menyimpan username dan password admin yang digunakan saat login admin.

Tabel 4.11 Perancangan Admin

|  |  |  |
| --- | --- | --- |
| **Field** | **Type** | **Keterangan** |
| id | Int(11) | *Primary Key*   |  |  | | --- | --- | | (*auto\_increment*) | [Change](http://localhost/phpmyadmin/tbl_structure.php?db=ket_surat&table=penduduk&token=c74aa92b6907c66973e02b8535d1bc83&goto=tbl_structure.php&back=tbl_structure.php&field=id&change_column=1) | |
| id\_petugas | Varchar(11) | Foreign key |
| username | Varchar(15) |  |
| password | Varchar(50) |  |
| nama | Varchar(50) |  |
| foto | Varchar(255) |  |
| role | Varchar(20) |  |
| level | Int(1) |  |
| RT | Int(2) |  |
| RW | Int(2) |  |

* + 1. **Perancangan Tabel Petugas**

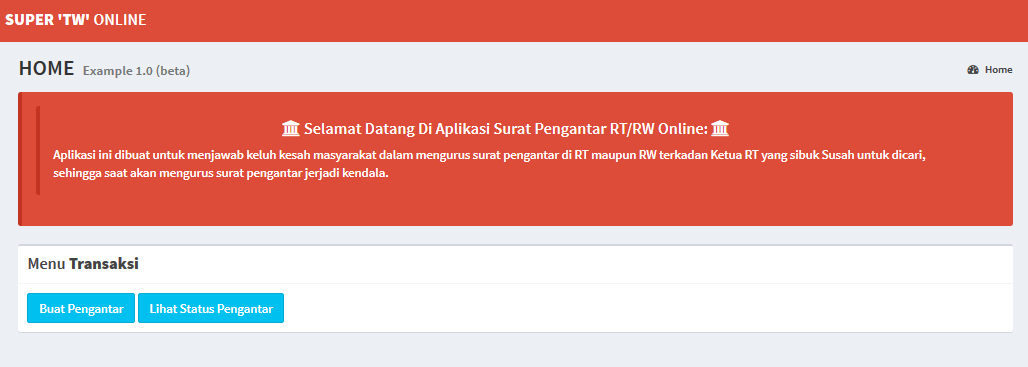
Tabel ini berfungsi untuk menyimpan informasi data petugas yang berhubungan dengan admin.

Tabel 4.12 Perancangan Tabel Peminjaman

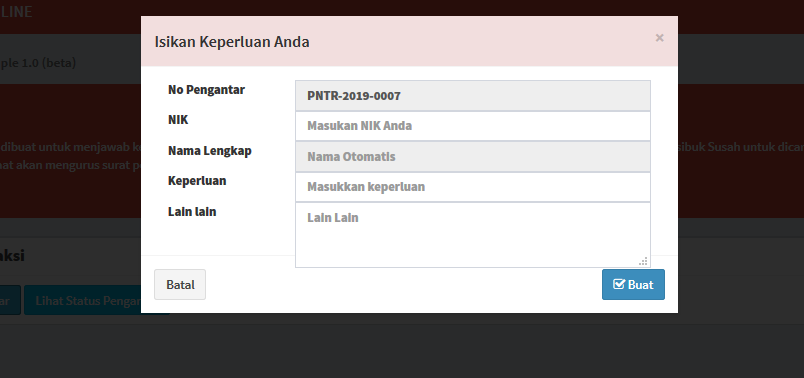
|  |  |  |
| --- | --- | --- |
| **Field** | **Type** | **Keterangan** |
| id\_petugas | Varchar(11) | *Primary Key* |
| nama | Varchar(30) |  |
| alamat | Varchar(50) |  |
| RT | int(2) |  |
| RW | Int(2) |  |
| no\_tel | Bigint (13) |  |
| email | Varchar(30) |  |
| jabatan | Varchar(10) |  |

* 1. **Tampilan Antarmuka Aplikasi**

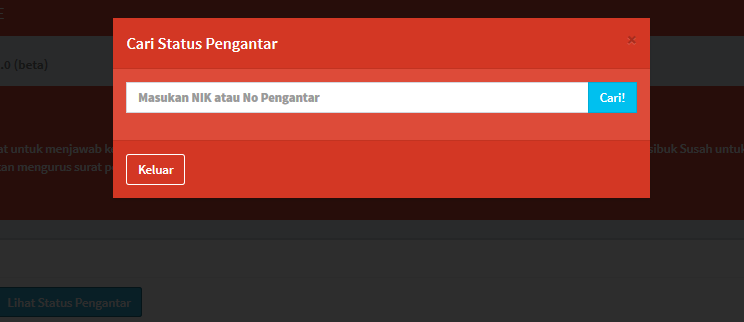
Tanpilan antarmuka aplikasi ini didesign sesederhana mungkin dan sesimple mungkin agar memudahkan warga unruk menggunakan tanpa basic pengetahuan tentang computer, dan tampilanya sebagai berikut:

1. Halaman Utama Pembuatan Pengantar

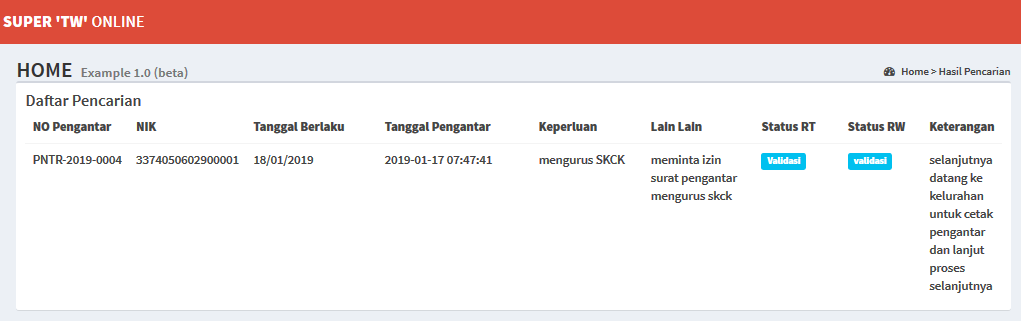
Gambar 4.20 Tampilan Halaman Utama Pengantar

1. Tampilan Form Buat Pengantar

Gambar 4.21 Tampilan Form Buat Pengantar

1. Tampilan Cari Status Pengantar

Gambar 4.22 Tampilan Cari Status Pengantar

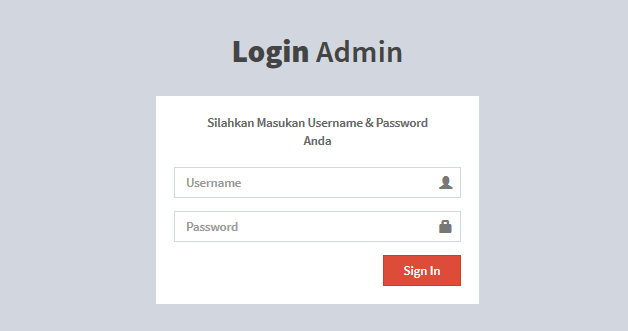
1. Hasil Pencarian Status Pengantar

Gambar 4.23 Tampilan Form Buat Pengantar

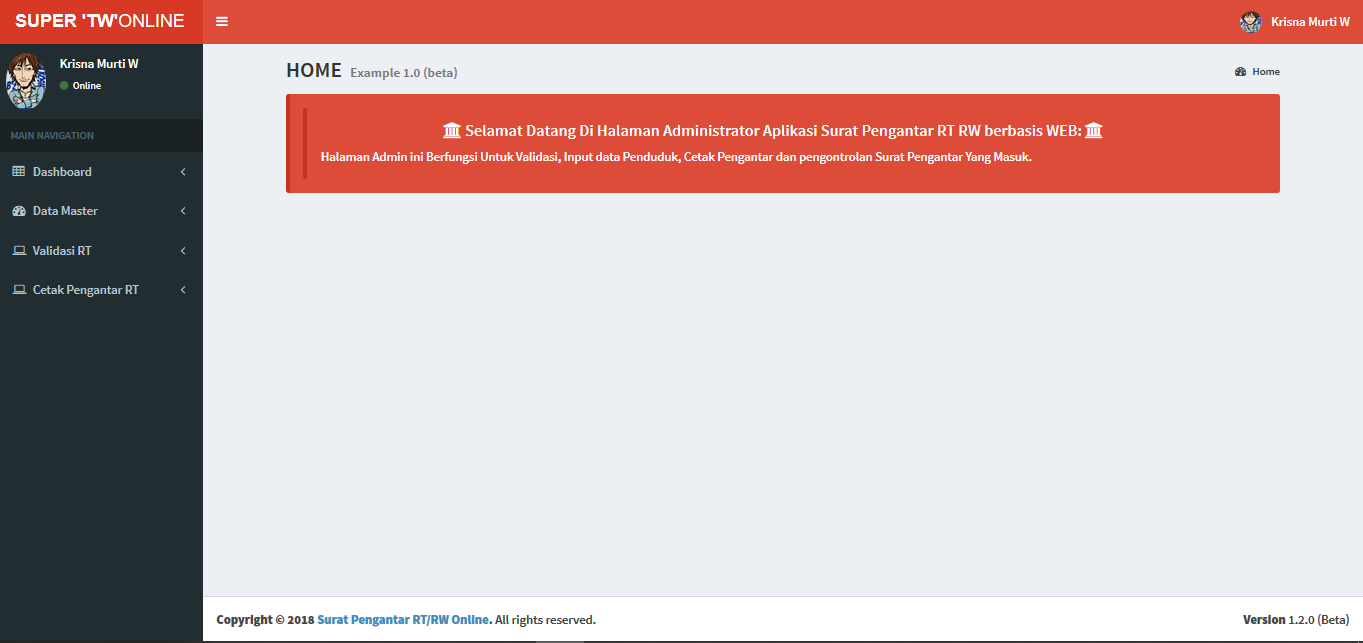
1. Tampilan Login Admin

Form login digunakan sebagai keamanan dari pihak yang tidak mempunyai hak akses. User diharuskan login terlebih dahulu sebelum masuk ke halaman menu. Di aplikasi pengantar Admin di bagi menjadi tiga level yaitu:

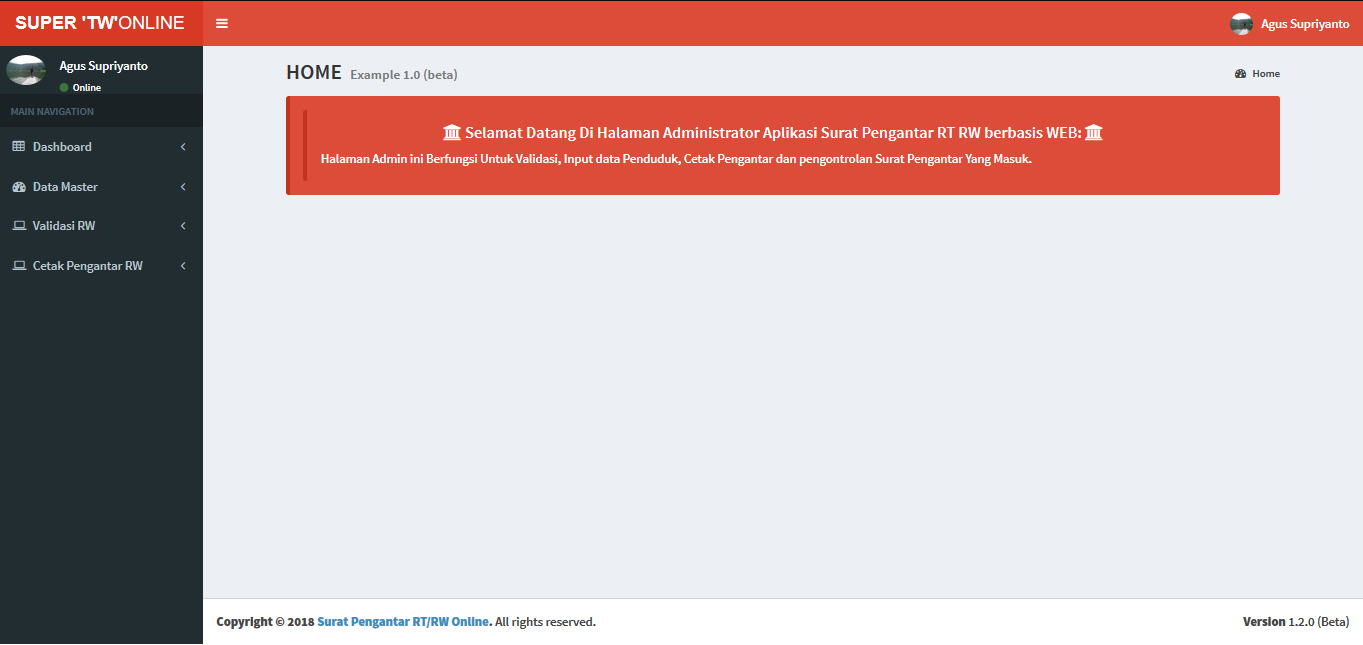
1. RT meliputi menu yang bisa di akses : dashboard, tambah data penduduk, validasi RT, Cetak pengantar
2. RW meliputi menu yang bisa di akses : dashboard, tambah data penduduk, validasi RW, Cetak pengantar
3. Kelurahan meliputi menu yang bisa di akses : dashboard, tambah data penduduk, validasi Kelurahan, Daftar Petugas



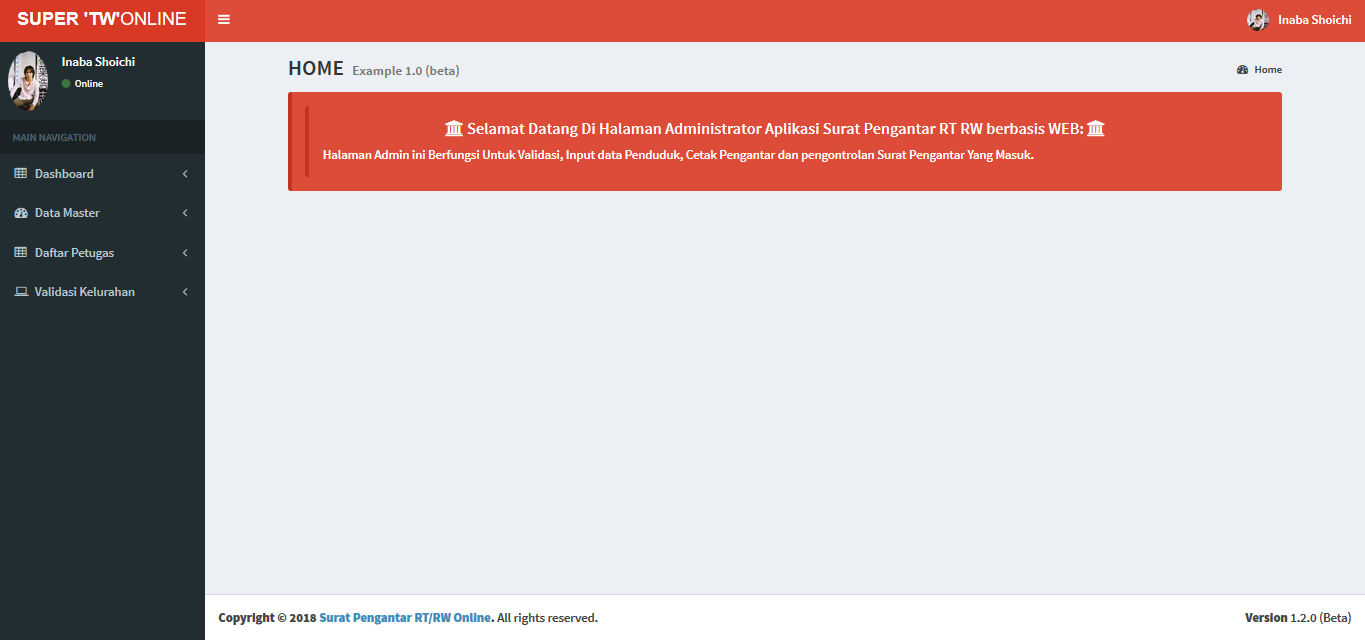
Gambar 4.24 Tampilan Halaman Login

1. Tampilan Dashboard Admin RT

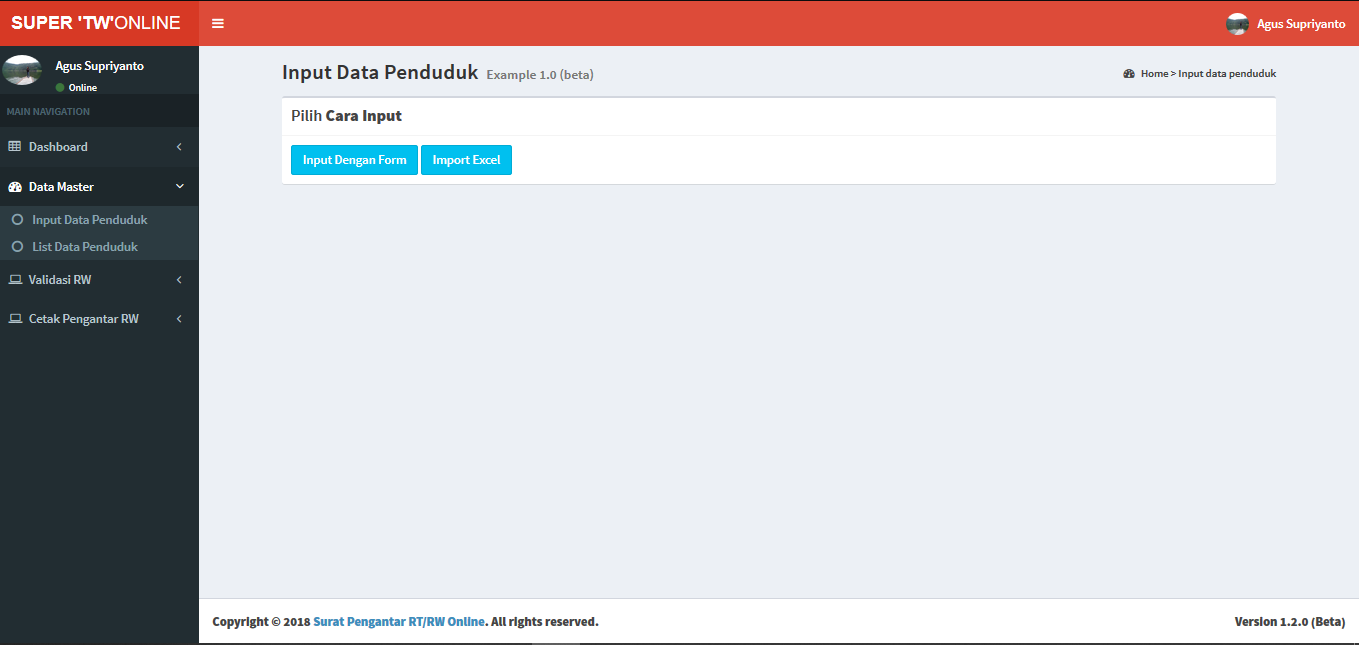
Gambar 4.25 Tampilan Dashboard Admin RT

1. Tampilan Dashboard Admin RW

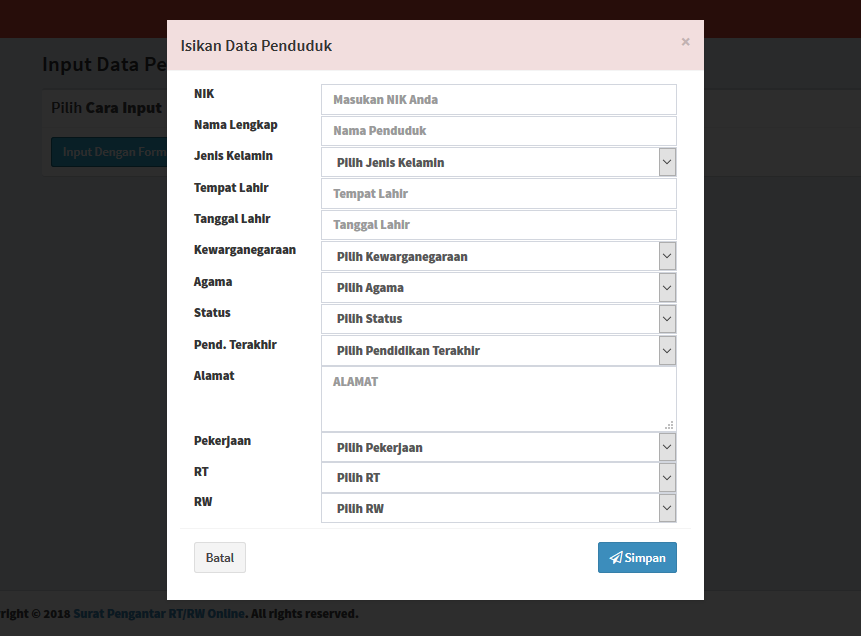
Gambar 4.26 Tampilan Dashboard Admin RW

1. Tampilan Dashboard Admin Kelurahan

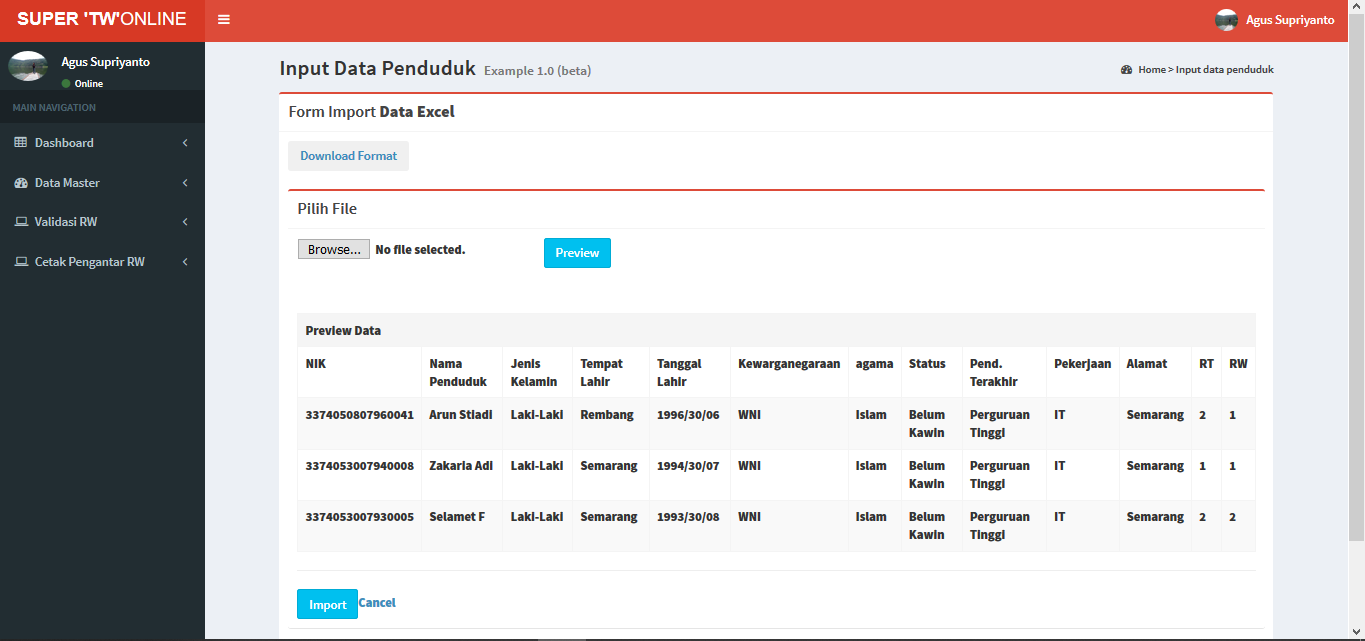
Gambar 4.27 Tampilan Dashboard Admin Kelurahan

1. Tampilan Menu Data Master

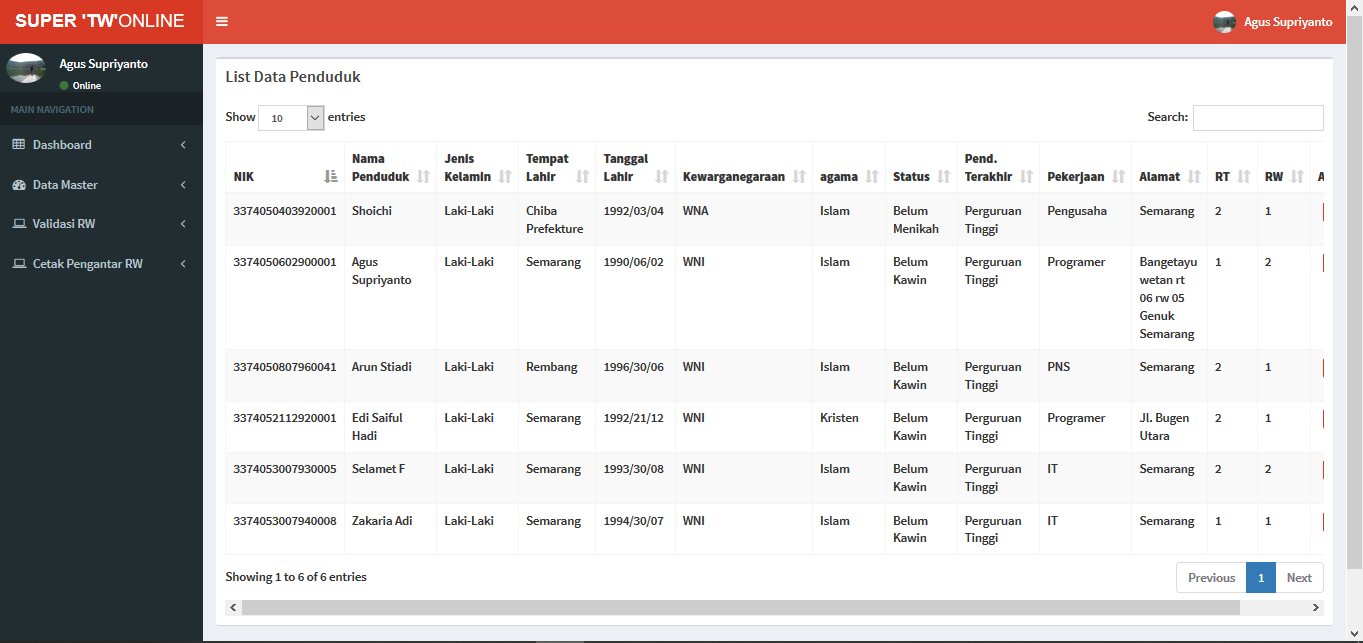
Gambar 4.28 Tampilan Menu Data Master Input Penduduk

1.  Tampilan Form Input Penduduk

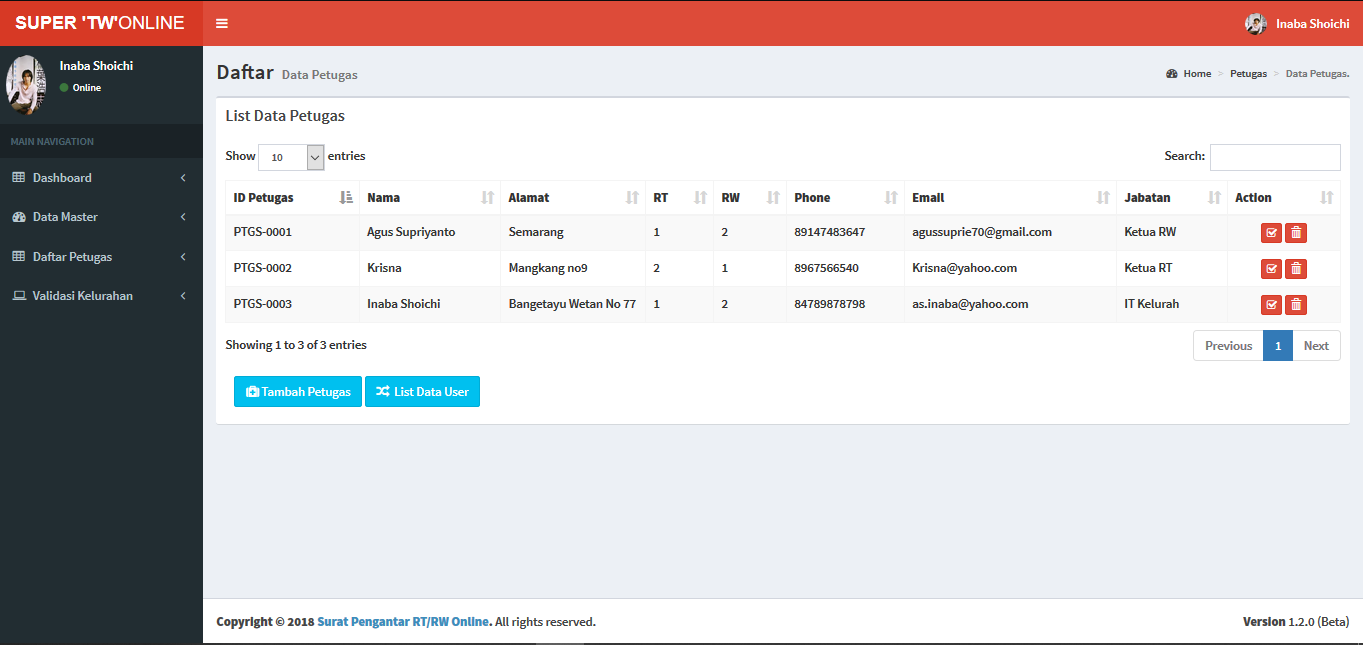
Gambar 4.29 Tampilan Form Input Penduduk

1. Tampilan Import Data Excel

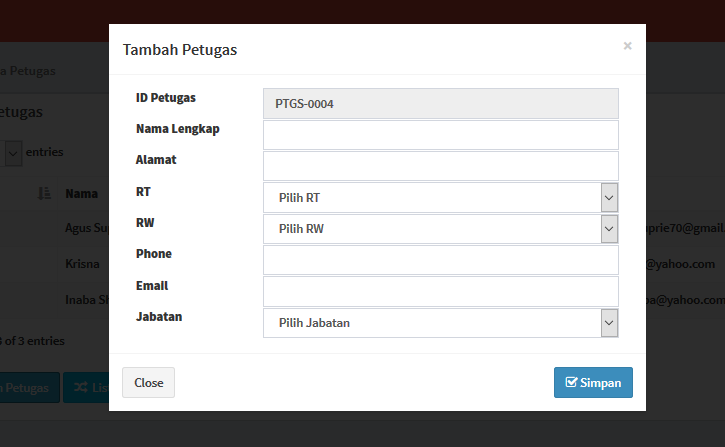
Gambar 4.30 Tampilan Import Data Excel

1. Tampilan List Data Penduduk

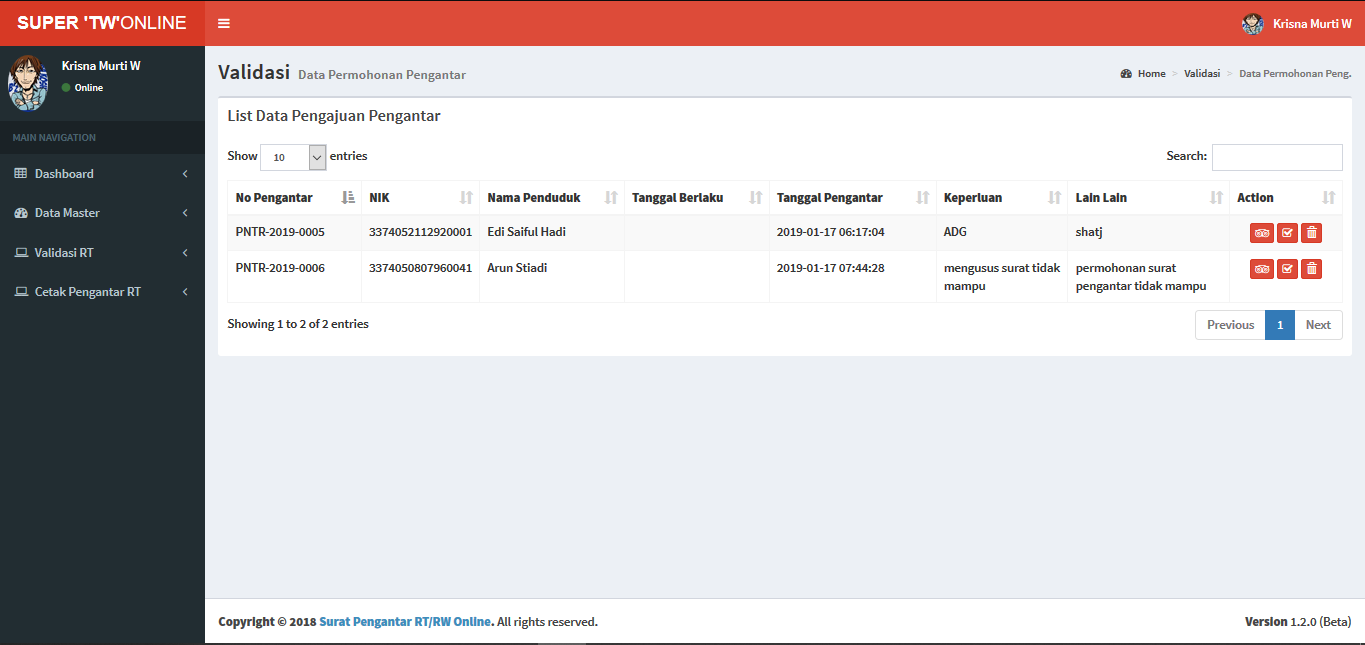
Gambar 4.31 Tampilan Import List Data Penduduk

1. Tampilan Menu Data Petugas

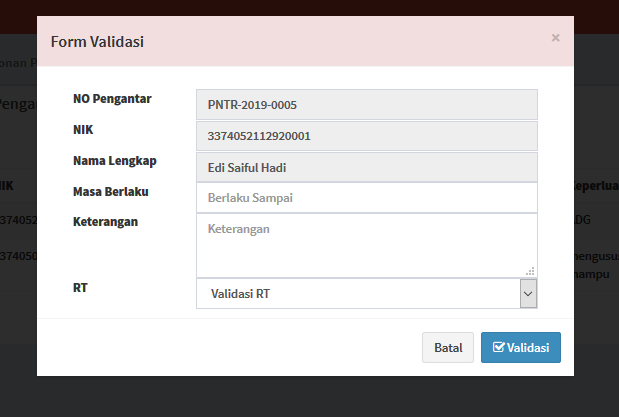
Gambar 4.32 Tampilan List Data Petugas

1. Tampilan Form Tambah Petugas

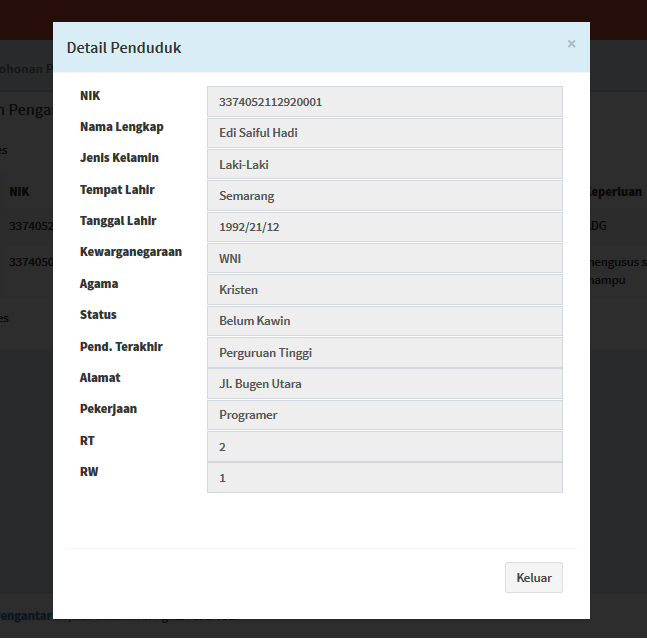
Gambar 4.33 Tampilan Form Tambah Petugas

1. Tampilan Menu Validasi

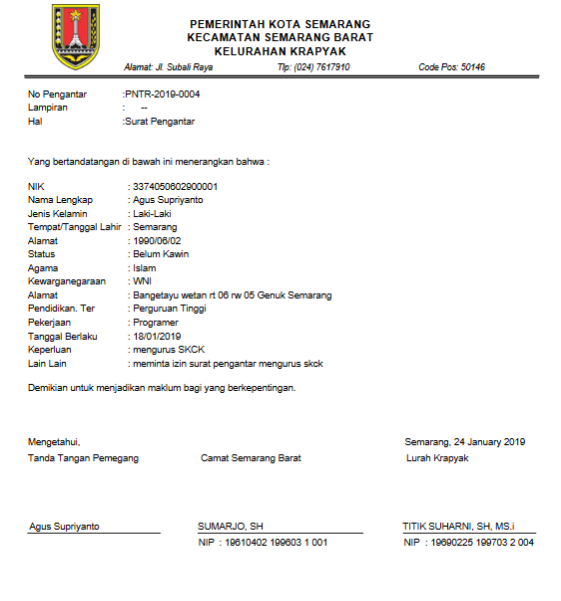
Gambar 4.34 Tampilan Menu Validasi

1. Tampilan Form Validasi

Gambar 4.35 Tampilan Form Validasi

1. Tampilan Detail Penduduk Yang Membuat Pengantar

Gambar 4.36 Tampilan Form Detail Penduduk Pembuat Pengantar

1. Hasil Jadi Surat Pengantar

Gambar 4.37 Tampilan Hasil Cetak Pengantar